IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-7, July 2018 E-ISSN: 2347-2693

Relevant Keyword Search for Building Service-Based System

G.B. Sai Preethi[1*], P. Radhika Raju[2], A. Ananda Rao[3]

[1*]M.Tech Scholar, Department of CSE, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India
[2] Ad-hoc Assistant Professor, Department of CSE, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India.
[3]Professor, Department of CSE, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India.

[*]*Corresponding Author: govasaipreethi11@gmaill.com, Tel.: +91-9908929010*

Available online at: www.ijcseonline.org

Received: 21/07/2018, Revised: /…./2018, Accepted: …/…/2018, Published: 31/July/2018

***Abstract*—** Software engineering has witnessing rapid changes from time to time. One such change of late is to have Service Oriented Architecture – based applications that can provide a comprehensive service to end users. Due to the emergence of web services technology with distributed, scalable, and interoperable computing capabilities, they are widely used to have service oriented applications. Such applications are known as Service Based Systems (SBSs). It is important to have automatic selection of composition of services at runtime to have dynamic integration of services. The problem with existing approaches is that they expect SOA techniques from software engineers and thus the quality of SBS depends on the expertise of the engineers. To overcome this problem, in this paper, introduced and implemented a framework that guides users to have certain keywords to search and build new SBS that improves Quality of Service (QoS) of generated SBSs. The service discovery and service composition are evaluated with a prototype application built to demonstrate proof of the concept. The experimental results revealed the usefulness of the proposed architecture which supports relevant keyword search for building SBS.

***Keywords*—**Keyword Search, Service Based System, Web Service, Service Discovery, Service Composition

I. INTRODUCTION

Service Oriented Architecture has become a major framework for building complex distributed software systems by discovering web services. Demand of building service-based system in SOA applications with the usage of composed available web services had increased. Most important step in service-based system development is to recognize the suitable services to the service-based system engineering process. The major contributions in this paper are proposing a framework known as Service Discovery and Service Composition (SDSC) Framework which guides the users to have service discovery and composition of SBS with required QoS. In this research the mechanism is to have ideal SBSs so as to find good keywords in making service discovery and composition. It makes the job of end users simple. Building a prototype application can helps to demonstrate proof of the concept. The experimental results with the proposed framework revealed the importance of the proposed application in service discovery and composition. The remainder of the paper is structured as follows. Section 2 provides review of literature on the web service anti-patterns and business process anti-patterns. Section 3 defined the problem considered. Section 4 presents the proposed system. Section 5 presents implementation details. Section 6 presents experimental results. Section 7 concludes the paper provides directions for future work.

II. RELATED WORK

This section reviews literature on the service composition and other service oriented applications. Web services composition with skyline services with expected QoS is explored in [1]. Web services are used to form processes. In [2] BPEL processes that are capable of self-supervising are studied. With respect to next generation web applications, the authors of [3] opine that they are services mashups. Semantic knowledge is playing an important role in the modern computing. In [4] it is used to have services discovery and composition of web services to form SBS. In [5] a framework is designed to have SBS with QoS-aware runtime composition of services. In this kind of scenario, query expansion can help to get better results. Query expansion techniques are explored in [6]. The concept of probabilistic match making is made in [7] for efficient service discovery and composition.

Network-awareness in the process of service composition in public cloud [8], search based approaches [9], monitoring strategies for SBS [10], situational web applications that are based on SOA [11], discovering web services in WWW [12], discovering services from relational databases with keyword search [13], service discovery for large scale networks [14] and search with top-k approach [15] are important works on the service composition and discovery. Wishful search [16], adaptive service composition [17], composition of

functionally diversified services [18], service recommendations with negative connection awareness [19] and keyword search mechanisms for building service based systems [20] are found in the literature with various approaches. A partnership based approach for improving QoS [21], interoperability based pervasive computing framework [22] are other important related work found.

Composite services with service ratings are explored in [23] while a framework is proposed in [24] for reusing services. Similar kind of reusability frameworks which are used to optimize program invariants [25,27]. The problem of imprecise service matching in the context of robust service composition is the study of [26] while an access control framework based on attributes and reputation is the focus in [28]. In the literature it is found that keyword based SBS building have issues as the users are not fully aware of perfect keywords. To overcome this problem, building a system in this paper which recommends keywords as well for improving robustness of composed SBS.

III. PROBLEM DEFINITION

Service Based Systems (SBSs) play vital role in rendering services to end users. They can get all related services with a single application instead of moving to different individual applications. When SBS is constructed dynamically, there is the problem of service discovery and service composition. From the literature it is understood that the service composition needs search keywords. When the users are not equipped with good keywords, it results in SBS composition that does not provide accurate solution. Therefore it is important to suggest keywords also to end users so as to help them to provide ideal keywords for search based SBS composition. This is the challenging problem considered in this paper.

IV. PROPOSED FRAMEWORK

The proposed work is a framework known as Service Discovery and Service Composition (SDSC) framework. The framework provides reusable building blocks to make the service discovery and service composition easier. Services related to different domains can be discovered dynamically and they are composed to form an SBS that renders required services to end users. The proposed framework is illustrated in Figure 1. There needs to be some planning for new SBS. The user needs to know functional requirements and constraints related to QoS. Then the services are discovered dynamically and then SBS is composed. As presented in Figure 1, it is evident that the services are discovered and SBS is composed based on search keywords. Therefore the framework performs service discovery to result in candidate services. From the candidate services, the final services are identified and composed. The problem identified here is that user may not be efficient or not equipped with the knowledge of keywords. Therefore the proposed is a mechanism to have the keyword extraction so that recommendations can be provided to end users.

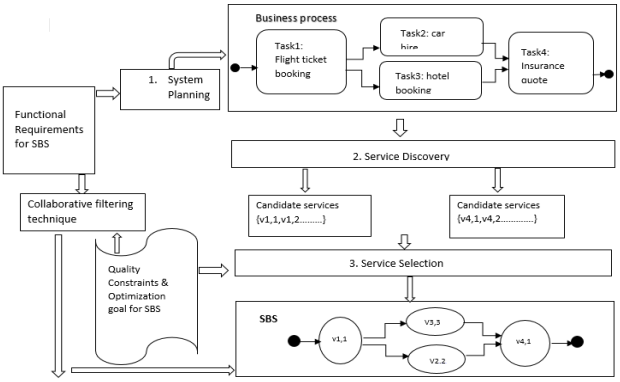

Figure 1. Proposed framework for SBS composition.

V. IMPLEMENTATION

The framework is implemented using Java programming language. Swing API is used to build intuitive Graphical User Interface (GUI). The dataset details are provided later in this section. The service discovery and service composition to form SBS are done automatically using the prototype application built. User performs loading datasets to have automated approach in discovering services and then perform SBS composition based on the keyword search mechanism to generate the SBS compositions. The keywords recommendations provided by the framework as illustrated and are used by end users.

5.1 Datasets Used

Datasets are collected from [29] for the experiments. Figure 2 shows an excerpt of dataset details. Dataset has services related to different domains like finance, science, shipping, travel and weather to mention few.

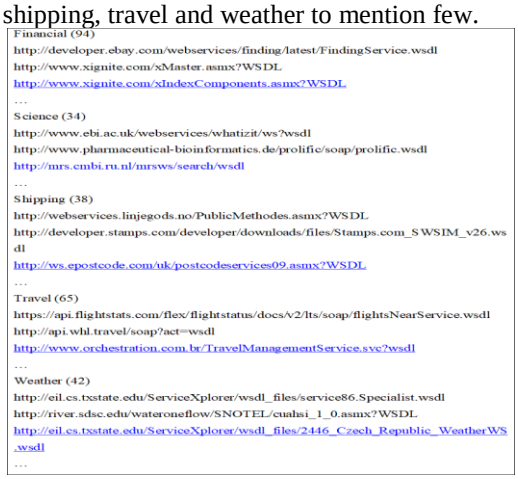
Financial (94)
http://developer.ebay.com/webservices/finding/latest/FindingService.wsdl
http://www.xignite.com/xMaster.asmx?WSDL
http://www.xignite.com/xIndexComponents.asmx?WSDL
...
Science (34)
http://www.ebi.ac.uk/webservices/whatizit/ws?wsdl
http://www.pharmaceutical-bioinformatics.de/prolific/soap/prolific.wsdl
http://mrs.cmbi.ru.nl/mrsws/search/wsdl
...
Shipping (38)
http://webservices.linjegods.no/PublicMethodes.asmx?WSDL
http://developer.stamps.com/developer/downloads/files/Stamps.com_SWSIM_v26.wsdl
http://ws.epostcode.com/uk/postcodeservices09.asmx?WSDL
...
Travel (65)
https://api.flightstats.com/flex/flightstatus/docs/v2/lts/soap/flightsNearService.wsdl
http://api.whl.travel/soap?act=wsdl
http://www.orchestration.com.br/TravelManagementService.svc?wsdl
...
Weather (42)
http://eil.cs.txstate.edu/ServiceXplorer/wsdl_files/service86.Specialist.wsdl
http://river.sdsc.edu/wateroneflow/SNOTEL/cualsi_1_0.asmx?WSDL
http://eil.cs.txstate.edu/ServiceXplorer/wsdl_files/2446_Czech_Republic_WeatherWS.wsdl
...

Figure 2. Dataset details

As shown in Figure 2, the dataset contains number of services categorized into different groups like finance, weather etc.

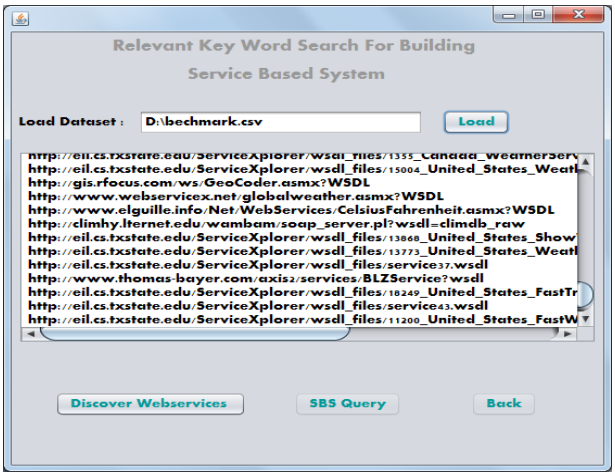

Figure 3. Shows main UI of the prototype

As shown in Figure 3, it is evident that it facilitates loading dataset and provides features like discovering web services; SBS query keyword recommendations and finally making SBS based on recommendations.

VI. EXPERIMENTAL RESULTS

Experiments are made with the prototype application using the dataset described in the previous section. The main observation is the computation time with respect to performance. However, the relevancy of the compositions are also observed and found to be effective in the proposed approach.

Table 1. Performance of the proposed system with different keyword distances.

Keyword Distance	**Computation Time(ms)**			
	KS3 Normal	*KS3 Constraint*	*KS3 Optimal*	*Proposed*
1	0	0	0	0
2	1000	1000	1000	800
3	1000	1000	7000	800
4	1000	1000	8000	800
5	1000	1000	10000	800
6	2000	2000	13000	1500
7	2000	2000	17000	1600
8	2500	2500	18000	2000
9	3000	4000	22000	2500
10	4000	5000	30000	3500

As shown in Table 1, it is evident that the computation time against different keyword distances is recorded with existing approaches like KS3 normal, KS3 constraint and KS3 optimal besides the proposed method.

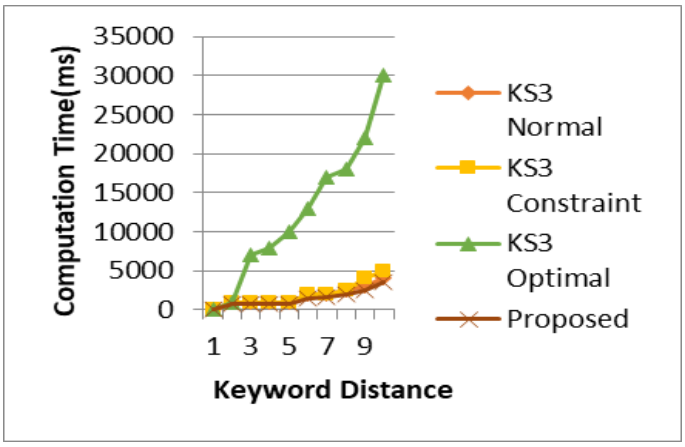

Figure 4. Performance comparison against different keyword distances

As presented in Figure 4, different number of keyword distances is presented in horizontal axis while the vertical axis showed computational time in milliseconds. The results revealed that the keyword distance has influence on the computational time.

Table 2: Performance of the proposed system with number of keywords in query.

Number of keywords in Query	**Computation Time(ms)**			
	KS3 Normal	**KS3 Constraint**	**KS3 Optimal**	**Proposed**
1	0	0	0	0
2	4000	4000	18000	3000
3	6000	12000	28000	5000
4	12000	25000	45000	10000

As shown in Table 2, it is evident that the computation time against different number of keywords in query is recorded with existing approaches like KS3 normal, KS3 constraint and KS3 optimal besides the proposed method.

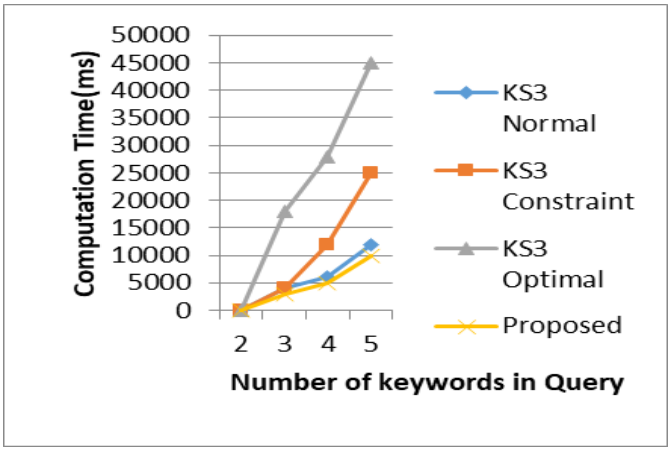

Figure 5. Performance comparison against different number of keywords in query

As presented in Figure 5, different number of keywords in query is presented in horizontal axis while the vertical axis showed computational time in milliseconds. The results revealed that the number of keywords in query has its influence on the computational time. Besides, it showed that the proposed system has better performance over existing system.

Table 3. Performance of the proposed system with different number of quality constraints.

Number of Quality Constraints	Computation Time(ms)			
	KS3 Normal	KS3 Constraint	KS3 Optimal	Proposed
1	1800	2500	2500	2200
2	1700	2300	2400	2000
3	2000	2000	2100	1800
4	2000	1600	1700	1400
5	2200	1500	1600	1300
6	1800	900	900	800
7	2000	500	500	400
8	1900	0	0	0
9	2100			
10	2000			

As shown in Table 3, it is evident that the computation time against different number of quality constraints is recorded with existing approaches like KS3 normal, KS3 constraint and KS3 optimal besides the proposed method.

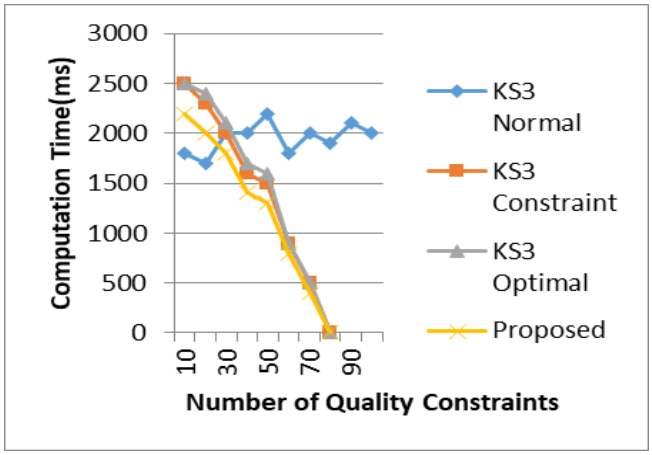

Figure 6: Performance comparison against different number of quality constraints.

As presented in Figure 6, different number of keywords in query is presented in horizontal axis while the vertical axis showed computational time in milliseconds. The results revealed that the number of keywords in query has its influence on the computational time. Besides, it showed that the proposed system has better performance over existing system.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, framework which is proposed to calculate the accurate and dynamic composition of SBS based on user query keywords. The process of identifying best SBS depends on the quality of service discovery and then selection. The problems involved in the dynamic SBS include that the process needs user involvement with knowledge on the keywords for efficient search based SBS composition. Other issues are related to performance of the system. In the literature these problems are found with the existing systems. In the proposed system the main framework named SDSC for service discovery and composition. Besides it helps users to gain recommendations to have better outcomes in the search based composition of SBS and building a prototype application to demonstrate proof of the concept. The experimental results proved that the proposed system showed better performance over existing system.

In future, studying of various algorithms in order to improve the possibility of using ensemble approaches to improve QoS in derived SBSs can be done.

REFERENCES

[1] M. Alrifai, D. Skoutas, and T. Risse, "Selecting Skyline Services for QoS-based Web Service Composition," *Proc of 19th International Conference on World Wide Web (WWW 2010)*, Raleigh, North Carolina, USA, pp. 11-20, 2010.

[2] L. Baresi and S. Guinea, "Self-Supervising BPEL Processes," *IEEE Transactions on Software Engineering*, vol. 37, no. 2, pp. 247-263, 2011.

[3] D. Benslimane, S. Dustdar, and A. Sheth, "Services Mashups: The New Generation of Web Applications," *IEEE Internet Computing*, vol. 12, no. 5, pp. 13-15, 2008.

[4] A. Brogi, S. Corfini, and R. Popescu, "Semantics-Based Composition-Oriented Discovery of Web Services," *ACM Transactions on Internet Technology*, vol. 8, no. 4, pp. 19:1-19:39, 2008.

[5] V. Cardellini, E. Casalicchio, V. Grassi, S. Lannucci, F. Lo Presti, and R. Mirandola, "MOSES: A Framework for QoS Driven Runtime Adaptation of Service-Oriented Systems," *IEEE Transactions on Software Engineering*, vol. 38, no. 5, pp. 1138-1159, 2012.

[6] C. Carpineto and G. Romano, "A Survey of Automatic Query Expansion in Information Retrieval," *ACM Computing Surveys*, vol. 44, no. 1, pp. 1-50, 2012

[7] G. Cassar, P. Barnaghi, and K. Moessner, "Probabilistic Matchmaking Methods for Automated Service Discovery," *IEEE Transactions on Services Computing (TSC)*, vol. 7, no. 4, pp. 654-666, 2014.

[8] A. Klein, F. Ishikawa, and S. Honiden, "Towards Network-Aware Service Composition in the Cloud," *Proc of 21st World Wide Web Conference (WWW 2012)*, Lyon, France, pp. 959-968, 2012.

[9] K. Golenberg, B. Kimelfeld, and Y. Sagiv, "Keyword Proximity Search in Complex Data Graphs," *Proc of 28th ACM SIGMOD International Conference on Management of Data (SIGMOD 2008)*, pp. 927-940, 2008.

[10] Q. He, J. Han, Y. Yang, H. Jin, J.-G. Schneider, and S. Versteeg, "Formulating Cost-Effective Monitoring Strategies for Services-based Systems," *IEEE Transactions on Software Engineering*, vol. 40, no. 5, pp. 461-482, 2014.

[11] X. Liu, Y. Ma, G. Huang, J. Zhao, H. Mei, and Y. Liu, "Data-Driven Composition for Service-Oriented Situational Web Applications," *IEEE Transactions on Services Computing (TSC)*, vol. 8, no. 1, pp. 2-16, 2015

[12] E. Al-Masri and Q. H. Mahmoud, "Investigating Web Services on the World Wide Web," *Proc of 17th International Conference on World Wide Web (WWW 2008)*, Beijing, China, pp. 795-804, 2008.

[13] V. Hristidis and Y. Papakonstantinou, "DISCOVERY: Keyword Search in Relational Databases," *Proc of 28th International Conference on Very Large Data Bases (VLDB 2002)*, Hong Kong, China, pp. 670-681, 2002.

[14] Ramya Swetha, Ananda Rao. A, Radhika Raju. P, Ramesh. G "Automation of Test Knowledge Reconfiguration for Web Services", South Asain Journal of Engineering and Technology, Vol. 2, No. 28(2016) 1-4.

[15] M. Jiang, A. W.-C. Fu, and R. C.-W. Wong, "Exact Top-k Nearest Keyword Search in Large Networks," *Proc of 36th ACM SIGMOD International Conference on Management of Data (SIGMOD 2015)*, pp. 393-404, 2015.

[16] A. V. Riabov, E. Boillet, M. D. Feblowitz, Z. Liu, and A. Ranganathan, "Wishful Search: Interactive Composition of Data Mashups," *Proc of 17th International Conference on World Wide Web (WWW 2008)*, Beijing, China, pp. 775-784, 2008.

[17] D. Ardagna and B. Pernici, "Adaptive Service Composition in Flexible Processes," *IEEE Transactions on Software Engineering (TSE)*, vol. 33, no. 6, pp. 369-384, 2007.

[18] F. Wagner, B. Klöpper, F. Ishikawa, and S. Honiden, "Towards Robust Service Compositions in the Context of Functionally Diverse Services," *Proc of 21st International World Wide Web Conference (WWW 2012)*, Lyon, France, pp. 969-978, 2012.

[19] Y. Ni, Y. Fan, W. Tan, K. Huang, J. Bi, "NCSR: Negative-connection-aware service recommendation for large sparse service network", *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 579-590, Apr. 2016.

[20] He et al., "*Keyword Search for Building Service-Based Systems*," in IEEE Transactions on Software Engineering, vol. 43, no. 7, pp. 658-674, 2017. doi:10.1109/TSE.2016.2624293

[21] Comi A., Fotia L., Messina F., Rosaci D., Sarné G.M.L. A partnership-based approach to improve QoS on federated computing infrastructures. Inf. Sci. 2016;367:246–258. doi: 10.1016/j.ins.2016.05.051.

[22] XWARE—A customizable interoperability framework for pervasive computing systems. Pervasive and Mobile Computing 47, 13-30. Online publication date: 1-Jul-2018.

[23] Enhance Trust Management in Composite Services with Indirect Ratings. The Computer Journal 60:11, 1619-1632. Online publication date: 1-Nov-2017.

[24] Towards a Reuse Strategic Decision Pattern Framework – from Theories to Practices. Information Systems Frontiers 13. Online publication date: 9-May-2018.

[25] P. Radhika Raju, Prof. A. Ananda Rao,, "*Optimization of Program Invariants*.", ACM SIGSOFT Software Engineering Notes, Vol. 39, Issue 1,Jan 2014.

[26] Imprecise Matching of Requirements Specifications for Software Services Using Fuzzy Logic. IEEE Transactions on Software Engineering 43:8, 739-759. Online publication date: 1-Aug-2017.

[27] Towards a Reuse Strategic Decision Pattern Framework – from Theories to Practices. Information Systems Frontiers 13. Online publication date: 9-May-2018.

[28] Dr. P. Radhika Raju, Prof. A. Ananda Rao, B. Swarna Jyothi, "*Development of Service Oriented Situational Web Applications Using Knowledge Discovery*.", International Journal of Advances Research in Computer Sciences and Software Engineering, Vol. 6-9, no.02, 2017.

[29] .Web services datasets. Retrieved from https://github.com/wsdream/wsdream-dataset. Accessed on 10 March 2018.

Authors Profile

Ms.G.B. Sai Preethi pursed Bachelor of Technology from Srinivasa Ramanujan Institute of Technology in 2016. She is pursuing Master of Technology in Computer Science and Engineering from Jawahralal Nehru Technological University, Ananthapuramu in year 2018, Andhra Pradesh, India.

P.Radhika Raju received B.Sc. (Comp. Sci) from Sri Krishnadevaraya University, Ananthapuramu; MCA degree from Indira Gandhi National Open University (IGNOU), India and M.Tech degree in Computer Science from Jawaharlal Nehru Technological University Anantapur, Ananthapuramu, A.P, India. She Completed her Ph.D degree from JNT University Anantapur. She authored a text

book and has publications in National and International Journals/Conferences. Her research interest include primarily Software Engineering. She is now working as a Lecturer in department of Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu, A.P.

Dr.A. Ananda rao received his B. Tech. Degree in computer science and engineering from University of Hyderabad, erstwhile Andhra Pradesh, India and M. Tech. Degree in A.I &Robotics from University of Hyderabad, erstwhile Andhra Pradesh, India. He received his Ph. D. Degree from Indian Institute of Technology Madras, Chennai, India. He is a professor of Computer Science & Engineering Department and currently working as a Director, Research & Development, JNT University Anantapur, Anantapuramu, Andhra Pradesh, India. Dr. Rao has published more than 160 publications in various national and international journals/conferences. He received the best research paper awards in 2009 and 2014. He also received best educationist award, Bharat VidyaShiromani Award, RashtriyaVidyaGaurav Gold Medal Award, Best Computer Teacher Award and Best Teacher Award from the Andhra Pradesh chief minister for the year 2014. His main research interest includes software engineering and Data mining. He is Senior Member IAENG.

